REMARKS

Claims 1-8, 10-15 and 17 are remaining in the present patent application. Claim 1 has been amended. No new matter has been added. Support for the amendment to Claim 1 can be found, among other places, in the instant application at original Claim 9 and lines 11-18 of page 6, which states,

> Service processor 120 is integrated as an input/output (I/O) device to server 110, and acts as an autonomous embedded device, which is powered independently and runs embedded applications independent of server 110's state. Server 110 may properly function with or without service processor 120 or with service processor 120 being inoperative. Further, service processor 120 is commercially available without a terminal, and is referred to as an embedded management processor or device because service processor 120 is part of server 110 and provides management services for server 110 (emphasis added).

Applicants respectfully submit that although Applicants amended Claim 1, the amendment would not cause the Examiner to perform another search. Therefore, should the following arguments be found to be persuasive, Applicants respectfully submit that it would be improper to make the next Office Action final due to the amendment in this response.

35 U.S.C. §102 Rejection

CLAIMS 10-14

In paragraph 6, the Office Action rejected Claims 10-14 under 35 U.S.C. 102(a) as being anticipated over U.S. patent publication no. 2003/0018681 by Subramanian et al. (referred to hereinafter as "Subramanian"). Applicants have reviewed Subramanian and respectfully submit that the embodiments recited by Claims 10-14 are not anticipated by the Subramanian.

Claim 10 recites,

A processing system comprising:

a first system;

an autonomous second system embedded in the first system;

an exception handler running in the second system for recording exception information upon an occurrence of an exception in the second system; and

a recovery agent running on the second system, for taking an action upon the occurrence of the exception based on the recorded exception information, wherein the action is performed outside of a debugging operation;

wherein the action corresponds to a type of exception that occurred in a programming task.

MPEP §2131 provides:

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). ... "The identical invention must be shown in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). The elements must be arranged as required by the claim.

SUBRAMANIAN

Subramanian states in paragraph 0002-0004, that there are certain types of exceptions in applications that result in the operating system gaining control and terminating the execution of the application. The first sentence of paragraph 0004 states, "Conventionally, applications do not have any mechanism that let themselves handle these exceptions programmatically." The first sentence of paragraph 0010 states, "An operating system has a top level exception handler which terminates an application as a default action upon receipt of any exceptions that occurred due to runtime problems of an application."

Referring to lines 9-13 of paragraph 0013, Applicants understand Subramanian to teach placing an exception trapper 32 between a top level exception handler 6 and an exception dispatcher 4. The top level exception

handler 6 and the exception dispatcher 4 are associated with an operating system and the exception trapper 32 is associated with an application recovery system 30/90. The exception trapper 32 enables an application to analyze an exception that occurs in the application and to prevent the operating system from terminating the application.

SUBRAMANIAN DOES NOT ANTICIPATE

Claim 10 recites, "…an occurance of an exception in the second system," and "an autonomous second system embedded in the first system." Subramanian's exceptions occur in Subramanian's application. Applicants do not understand Subramanian's application to be an autonomous second system embedded in a first system. At a minimum, Applicants understand Subramanian's application to depend on Subramanian's operating system and to depend on the computer system that Subramanian's application executes on. Therefore, Applicants respectfully submit that Subramanian does not teach, describe, or suggest, among other things, "an autonomous second system embedded in the first system," as recited by independent Claim 10.

SUMMARY FOR 102

Since Subramanian does not teach, describe or suggest "an autonomous second system embedded in the first system," as recited by independent Claim 10 Applicants respectfully submit that Claim 10 is not anticipated by Subramanian. Claims 11-14 depend on Claim 10 and include all of the features of independent Claim 10. Therefore, these dependent claims should be patentable for at least the reasons that Claim 10 should be patentable.

35 U.S.C. §103 Rejection

CLAIMS 1-8, 15 and 17

In paragraph 4, the Office Action rejected Claims 1-8, 15 and 17 under 35 U.S.C. 103(a) as being unpatentable over Subramanian in view of U.S. patent

publication no. 2002/0029299 by Kappel et al. (referred to hereinafter as "Kappel"). Applicants have reviewed Subramanian and Kappel and respectfully submit that the embodiments recited by Claims 1-8, 15 and 17 are patentable over the Subramanian or Kappel, alone or in combination.

Claim 1 recites,

> An exception handling mechanism comprising:
> an exception handler for recording exception information dependant on types of exceptions and programming tasks that encounter exceptions; and
> a recovery agent for taking an action upon an occurrence of an exception that occurred for a programming task, wherein the action is performed outside of a debugging operation;
> wherein the action to be taken upon the occurrence of the exception corresponds to a type of exception and a programming task, and includes one or a combination of restarting the programming task, terminating the programming task, resetting a system running the programming task, and disregarding the exception,
> wherein the exception handler and the recovery agent run on a first system that operates autonomously and the first system is embedded in a second system.

Applicants respectfully submit that "[i]t is improper to combine references where the references teach away from their combination" (emphasis added; MPEP 2145(X)(D)(2); *In re Grasselli*, 713 F.2d 731, 743, 218 USPQ 769, 779 (Fed. Cir. 1983)). Applicants respectfully note that "[a] prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention" (emphasis in original; MPEP 2141.02(VI); *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984)). Further, Applicants respectfully submit that, "[w]ith regard to rejections under 35 U.S.C. 103, the examiner must provide evidence which as a whole shows that the legal determination sought to be proved (i.e., the reference teachings establish a *prima facie* case of obviousness) is more probable than not" (emphasis added) (MPEP 2142).

In particular, "[i]f the proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed amendment" (emphasis added) (MPEP 2143.01(V); *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984)).

More specifically, Applicants respectfully submit that there is no motivation to combine the teachings of Subramanian and Kappel, because these references teach away from the suggested modification. For example, Applicants understand the combination of Subramanian and Kappel to render each others' teachings unsatisfactory for their intended purposes as will be described in more detail.

SUBRAMANIAN

Subramanian states in paragraph 0002-0004, that there are certain types of exceptions in applications that result in the operating system gaining control and terminating the execution of the application. The first sentence of paragraph 0004 states, "Conventionally, applications do not have any mechanism that let themselves handle these exceptions programmatically." The first sentence of paragraph 0010 states, "An operating system has a top level exception handler which terminates an application as a default action upon receipt of any exceptions occurred due to runtime problems of an application."

Referring to lines 9-13 of paragraph 0013, Applicants understand Subramanian to teach placing an exception trapper 32 between a top level exception handler 6 and an exception dispatcher 4. The top level exception handler 6 and the exception dispatcher 4 are associated with an operating system and the exception trapper 32 is associated with an application recovery system 30/90. The exception trapper 32 enables the application it is associated

with to analyze an exception that occurs in the application and to prevent the operating system from terminating the application.

Referring to the first sentence of paragraph 0046 and paragraph 0052, a Win32 function called SetUnhandledExceptionFilter can be used to cause a specified exception trapper 32 to gain control when an exception occurs instead of the top level exception handler 6. Referring to paragraphs 0055-0058, the application recovery system that an exception trapper 32 is a part of can be implemented as a structured exception handling (SEH) block. The first sentence of 0057 states, "The SEH block traps the exception on the message pump level." Paragraph 0058 states, "The SEH block overrides CwinApp::ProcessWndProException with a callback that allows the application to determine if any special conditions need to be met before terminating a message…"

KAPPEL

Kappel teaches a method for exception handling. Referring to the abstract, "the system includes a class creator that establishes a plurality of classes of exception types and an exception capture mechanism that captures an exception." Referring to the last sentence of paragraph 0003, Kappel's goal is to "…provide for a generalized or simplistic view of exception or error handling." Referring to the first sentence of paragraph 0029, Kappel achieves his goal by utilizing three generic exception types. Kappel states in the second to last sentence of paragraph 0029 that the exception types are application exceptions, system exceptions and validation exceptions.

THE COMBINATION OF SUBRAMANIAN AND KAPPEL

Subramanian teaches handling special conditions. For example, paragraph 0058 states, "The SEH block overrides CwinApp::ProcessWndProException with a callback that allows the application to determine if any special conditions need to be met before terminating a

message…" In contrast, referring to paragraph 0029 Kappel teaches "…the present invention utilizes an instance strategy pattern utilizing three generic exception types. …" Kappel goes so far as to say that "…an unlimited number of exceptions can be defined" however, "only three exceptions must be caught" (emphasis added). Therefore, modifying Kappel with Subramanian would render Kappel's results inoperable for Kappel's intended purposes. Further, modifying Subramanian with Kappel would render Subramanian's results inoperable for Subramanian's intended purposes.

SUMMARY FOR 103

Since modifying Subramanian with Kappel would render Subramanian's results inoperable for Subramanian's purposes and vice versa, Applicants respectfully submit that Subramanian and Kappel cannot be combined to teach, describe, or render obvious the embodiments recited by Claims 1-8, 15 and 17.

CONCLUSION

In light of the facts and arguments presented herein, Applicants respectfully request reconsideration of the rejected Claims.

Based on the arguments presented above, Applicants respectfully assert that Claims 1-8, 10-15 and 17 overcome the rejections of record. Therefore, Applicants respectfully solicit allowance of these Claims.

The Examiner is invited to contact Applicants' undersigned representative if the Examiner believes such action would expedite resolution of the present Application.

Respectfully submitted,

Wagner Blecher LLP

Date: 05/12/2008

/John P. Wagner, Jr./
John P. Wagner Jr.
Registration No. 35,398

Address: Westridge Business Park
123 Westridge Drive
Watsonville, California 95076 USA

Telephone: (408) 377-0500 Voice
(408) 234-3649 Direct/Cell
(831) 722-2350 Facsimile